APPENDIX A: ROM BIOS LISTINGS

Appendix A

```
LOC OBJ                LINE    SOURCE

E936 0D
E937 FF
E938 4153444647484A
     4B4C3A22
E943 7E                                DB      07EH,-1,'|ZXCVBNM<>?',-1,0,-1,' ',-1
E944 FF
E945 7C5A584356424E
     4D3C3E3F
E950 FF
E951 00
E952 FF
E953 20
E954 FF
                               ;----- UC TABLE SCAN
E955                           K12     LABEL   BYTE
E955 54                                DB      84,85,86,87,88,89,90
E956 55
E957 56
E958 57
E959 58
E95A 59
E95B 5A
E95C 5B                                DB      91,92,93
E95D 5C
E95E 5D
                               ;----- ALT TABLE SCAN
E95F                           K13     LABEL   BYTE
E95F 68                                DB      104,105,106,107,108
E960 69
E961 6A
E962 6B
E963 6C
E964 6D                                DB      109,110,111,112,113
E965 6E
E966 6F
E967 70
E968 71
                               ;----- NUM STATE TABLE
E969                           K14     LABEL   BYTE
E969 3738392D343536                    DB      '789-456+1230.'
     2B313233302E
                               ;----- BASE CASE TABLE
E976                           K15     LABEL   BYTE
E976 47                                DB      71,72,73,-1,75,-1,77
E977 48
E978 49
E979 FF
E97A 4B
E97B FF
E97C 4D
E97D FF                                DB      -1,79,80,81,82,83
E97E 4F
E97F 50
E980 51
E981 52
E982 53

                               ;----- KEYBOARD INTERRUPT ROUTINE

E987                                   ORG     0E987H
E987                           KB_INT  PROC    FAR
E987 FB                                STI                     ; ALLOW FURTHER INTERRUPTS
E988 50                                PUSH    AX
E989 53                                PUSH    BX
E98A 51                                PUSH    CX
E98B 52                                PUSH    DX
E98C 56                                PUSH    SI
E98D 57                                PUSH    DI
E98E 1E                                PUSH    DS
E98F 06                                PUSH    ES
E990 FC                                CLD                     ; FORWARD DIRECTION
E991 E8AA15                            CALL    DDS
E994 E460                              IN      AL,KB_DATA      ; READ IN THE CHARACTER
E996 50                                PUSH    AX              ; SAVE IT
E997 E461                              IN      AL,KB_CTL       ; GET THE CONTROL PORT
E999 8AE0                              MOV     AH,AL           ; SAVE VALUE
E99B 0C80                              OR      AL,80H          ; RESET BIT FOR KEYBOARD
```

Appendix A

Attach To Paper No. 5

Computer Dictionary and Handbook

by

Charles J. Sippl

and

Roger J. Sippl

Howard W. Sams & Co., Inc.
4300 WEST 62ND ST. INDIANAPOLIS, INDIANA 46268 USA

Indianapolis, Indiana 46268

THIRD EDITION
SECOND PRINTING—1980

International Standard Book Number: 0-672-21632-9
Library of Congress Catalog Card Number: 79-67133

Printed in the United States of America.

...rier locates the equipment which interconnects subscribers and circuits.

central processing element (bit slices) — Each central processing element (CPE) represents a 2-bit or 4-bit slice through the data-processing section of a computer. In some systems, several CPEs may be arrayed in parallel to form a processor of any desired word length. The microprocessor, which together with the microprogram memory controls the step-by-step operation of the processor, is itself a powerful microprogrammed state sequencer.

central processing unit — Abbreviated CPU. The unit of a computing system that contains the circuits that control and perform the execution of instructions.

central processing, unit loop — The main routine or a control program and that which is associated with the control of the internal status of the processing unit, in contrast to those control programs of routines developed with terminals and file storage input-output.

central processing unit, microcomputer — The CPU is the primary functioning unit of any computer system. Its basic architecture consists of storage elements called registers, computational circuits designated as the arithmetic-logic unit (ALU), the control block, and input-output ports. A microprocessor built with LSI technology often contains a CPU on a single chip. Because such a chip has limited storage space, memory implementation is added in modular fashion on associated chips. Most microcomputers consist of a CPU chip and others for memory and i/o.

central processor — *See* central processing unit.

central processor organization — The computer can be divided into three main sections: arithmetic and control, input/output, and memory. The arithmetic and control section carries out the directives of the program. The calculations, routing of information, and control of the other sections occurs in this part of the central processor. All information going in and coming out of the central processor is handled by the input/output section. It also controls the operation of all peripheral equipment. The memory section is the heart of the central processor; it provides temporary storage for data and instructions. Because of its importance, the total cycle time of the memory is the main determining factor in the overall speed of the processor.

central scanning loop — A loop of instructions which determines which task is to be performed next. After each item of work is completed, control is transferred to the central scanning loop which searches for processing requests in order to determine the next item to be processed. The computer may cycle idly in the central scanning loop if no item requires its attention, or it may go into a wait state which is interrupted if the need arises. The central scanning loop is the nucleus of a set of supervisory programs.

central terminal unit — Abbreviated CTU. This unit supervises communication between the teller consoles and the processing center. It receives incoming messages at random intervals, stores them until the central processor is ready to process them, and returns the processed replies to the teller consoles which originated the transactions (bank application).

cerdip — Abbreviation for Ceramic Dual In-line Package.

certified tape — Computer tape that is machine checked on all tracks throughout each roll and is certified by the supplier to have less than a specific total number of errors or to have zero errors.

certifier, tape — A peripheral device or unit designed to locate defects in magnetic tape before use, such as oxide emissions, unevenness, bubbles, etc.

CF — *See* control footing.

CH — *See* control heading.

chain — 1. Any series of items linked together. 2. Pertaining to a routine consisting of segments which are run through the computer in tandem, only one segment being within the computer at any one time and each segment using the output from the previous program as its input.

chain additions program — An instruction set that will permit new records to be added to a file.

chain, binary — A series of flip-flops (binary circuits) which exist in either one of two states, but each circuit can affect or change the following circuit.

chain code — An arrangement in a cyclic sequence of some or all of the possible

preparation of statistical and analytical reports dealing with the frequency of certain transactions.

log, remote computing-system — The remote computing system maintains a log of operations that take place between the computer and each terminal. The log contains such information as the number of statements handled, the number and types of errors detected, and the volume of output produced. The information in the log can be used for various purposes. For example, the number of errors may indicate that additional training might be helpful. Similarly, if an individual terminal is busy, it might indicate the need for an additional terminal. If the cost of the system is shared among terminals according to usage, the information in the log can be used for billing purposes.

log, system — A data set in which job-related information, operational data, descriptions of unusual occurrences, commands, and messages to or from the operator may be stored. Abbreviated SYSLOG.

log word, interrupt — *See* interrupt log word.

longitudinal check — A system of error control based on the check that some preset rules for the formation of the group of bits in the same numerical order in all the character signals in a block are observed.

longitudinal circuit — A circuit formed by one telephone wire (or by two or more telephone wires in parallel) with the return through the earth or through any other conductors except those which are taken with the original wire or wires to form a metallic telephone circuit.

longitudinal parity check — The data line terminal at the transmitting end generates a longitudinal parity character during the transmission of the data characters. This is essentially a count for even parity of all of the bits in each one of the bit levels for all data characters in the message including the start-of-message code but not the end-of-message code. This same count is also being generated for the bits of the data characters entering the data-line terminal of the receiving end.

longitudinal redundance — A condition in which the bits in each track or row of a record do not total an even (or odd) number. The term is usually used to refer to records on magnetic tape ... a system can have either odd or ... longitudinal parity.

longitudinal redundancy check — ... breviated LRC. A system of ... checking for transmission errors ... organized into blocks has a block ... or LRC character following the block. The LRC character is developed ... forming a parity check on all bits in ... same bit position in the block. This ... equivalent to forming a parity bit for ... bits located on each "channel" ... "level" of paper or magnetic tape.

longitudinal transmission check — An even or odd parity check at fixed intervals during data transmission.

long word — *See* word, long.

look ahead — A feature of the CPU that permits the machine to mask an interrupt request until the following instruction has been completed. This is also a feature of adder circuits and ALUs that permits these devices to look ahead to anticipate that all carries generated are available for addition.

look-at table — Finding elements of a table by direct calculation rather than by a comparison search.

look-up — A procedure for obtaining the function value corresponding to an argument from a table of function values.

look-up instruction — An instruction designed to allow reference to systematically arranged, stored data.

look-up table — A collection of data in a form suitable for ready reference, frequently as stored in sequenced machine locations or written in the form of an array of rows and columns for easy entry, and in which an intersection of labeled rows and columns serves to locate a specific piece of data or information.

look-up, table, instruction — *See* instruction, table-look-up.

look-up, table, techniques — *See* table look-up techniques.

loop — 1. The repeated execution of a series of instructions for a fixed number of times. 2. A coding technique in which a group of instructions is repeated, usually with modified instructions or modified data values. 3. A sequence of instructions that is repeated until a terminal condition prevails.

loopback test — A type of test in which signals are looped from a test center

memory only. In addition, often the main PROM has all the control lines available for implementing RWM (read/write memory) program memory. In small systems ROM program memory is used for systems in fixed applications. RWM memory is used where it is desired to change the system application by the operator. RWM is a considerable step in small system complexity in hardware and programs.

read/write scatter — An operation performed under program control that reads a block of data from tape and breaks it up into processable elements. After processing, data is recombined and written on the tape as a block.

ready — The status or condition of being ready to run. A program, task, or hardware device that is in a ready condition needs only a start signal in order to begin operation.

ready condition — A specification or circumstance of a job or task signified when all of its requirements for execution other than control of the central processor have been satisfied.

ready light — An indicator light on the display panel which, when on, indicates the equipment is ready for operation.

ready mode, time sharing — *See* time sharing, ready mode.

ready-record — A specific signal from a file-access mechanism to the computer that a record whose address was previously provided by a seek command has now been located and may be read into memory.

ready status word — A particular status word indicating that the remote computing system is waiting for entry from the terminal.

real constants — A real constant is written with a decimal point, using the decimal digits 0, 1, . . . , 9. A preceding + or − sign is optional. An unsigned real constant is assumed to be positive. An integer exponent preceded by an E may follow a real constant. The exponent may have a preceding + or − sign. An unsigned exponent is assumed to be positive.

real number — An element of a set of all positive and negative numbers, including all types: integers, zeros, mixed, rational, irrational, etc., but not imaginary or complex.

real ratio (time) — One computer time ratio is the time interval between two events in a simulation by a computer to the problem time, or the physical system time, i.e., the time interval between corresponding events in the physical system being simulated. When this ratio is greater than 1, the operation is considered to be on an extended time scale, which is a slow-time scale. When it is less than 1 it is said to be on a fast-time scale, and when it is not constant during the run it is said to be on a variable time scale. Real-time working is involved when it is equal to 1.

real time — 1. In solving a problem, a speed sufficient to give an answer within the actual time the problem must be solved. 2. Pertaining to the performance of a computation during the actual time that the related physical process transpires in order that results of the computation can be used in guiding the physical process.

real-time address — *Same as* address, immediate.

real-time addressing — *Same as* addressing, immediate.

real-time application — Real-time processing is accomplished on a time-current basis. It handles the flow of data from widespread manufacturing inventories and production lines such as the shifting pattern of transportation schedules, or the scattered operations of the utility industry. For example, in airlines reservation control, the real-time system provides an instantaneous picture of seat availability, cancellations, sales, and flight data for the whole airline. The airline agent simply presses buttons.

real-time, batch processing — The requirements for real-time action are known frequently to occur in peaks and valleys. In many businesses these requirements tend to increase from early morning through the middle of the day and to taper off from then on. In other businesses the occurrence of these demands may be sporadic. The real-time system is so designed that it will automatically, as its facilities are freed from the dynamic demands of real-time processing, load them up with the ordinary day to day backlog of less urgent work of the familiar batch-processing type—typically, the sequential processing of sequentially ordered files such as accounts receivable, payable, or payrolls.

FORTRAN IV uses SQRT (), eliminating the F. EXPF () in II, EXP () in IV to exponentiate to a power, i.e., (2).

13. A READ statement is used to enter stored data into the computer.
    (a) The READ statement indicates input or output operation, which variables are to receive new values, and the order of the values: "READ n, list" where n is the statement number, and list represents the number of variables to be read (or printed, punched, etc.).
14. The FORMAT statement, which must have a statement number, describes:
    (a) One punched card (older systems).
    (b) The specification of mode for each variable on the list.
        (1) I is used for integer mode variables.
        (2) F is used for floating-point variables punched in literal notation.
        (3) E is used for floating-point variables punched in exponential notation.
        (4) X is used to skip columns.
        (5) H is used to describe Hollerith fields.
    (c) The number of columns on the punched card which must be read for each specification, i.e., the field width.
    (d) The number of digits following the decimal point for F and E specifications.
    EXAMPLE (100 is the statment number)
    100 FØRMAT (I4, I3, F5.2, F9.7, F9.6)

    In addition:
    (e) There is an abbreviated notation for successive identical fields.
    FØRMAT (I4, I4, I4) = FØRMAT (3I4)
    (f) The decimal point does not need to be punched. It is sufficient to locate the decimal in the FØRMAT statement. If there is disagreement between the location of the decimal specified in the FØRMAT statement and the decimal actually punched on the card, the punched decimal takes precedence and is used.
    (g) Because FØRMAT statements are not executed, there are no restrictions on their location in the source program. It is good programming to write all the FØRMAT statements on a separate coding sheet and place the FØRMAT cards at the first part of the program.
15. The PAUSE and STOP statements allow the programmer to check interim results.
16. The END statement signals the completion of the source program and tells the computer to execute the object program.
17. In order to transfer control to a statement out of sequence, a GO TO statement is often used: $(n_1, n_2, n_3, \ldots, n_m)$ if the value of the integer variable is 1, control will go to the first statement number listed; i.e., n_1, if 2, to n_2, etc.
18. The IF statement transfers control on the condition of the happening of a certain event: IF $(A - B)$ n_1, n_2, n_3. If the result of $(A - B)$ is negative, control goes to statement n_1, if 0, to n_2, if positive, to n_3.
19. Subscribed variables (for arrays) allow the programmer to represent a number of variables with one name.
    (a) Individual variable subscripts are called elements.
    (b) The entire set of subscripts is called an array.
    (c) Fixed and floating-point variables must not be mixed in an array.
    (d) There are a number of rules for using subscripted variables.
        (1) You must tell the computer which variables are subscripted
        (2) How many elements are there in each array, and
        (3) How many subscripts are there for each subscripted variable.
        (4) Subscript cannot be floating-point, more than 3, or precede the DIMENSION statement.

Subscripted variables (single dimension) can represent any element of a one-dimensional array or table of numbers. The variable is still a FORTRAN variable of integer or floating-point mode, depending upon its first letter.

The FORTRAN statements illustrating the set of DØ in a counting loop below read one X-value at a time. The whole set of X-values can be thought of as a one-dimensional array or table,

$$X_1, X_2, X_3, \ldots, X_i, \ldots, X_N$$

FORTRAN provides a means to represent any element of a one-dimensional (and 2 or 3 dimensional) array by appending one subscript to the variable. For example, the variable X_i can be written X (I), a FORTRAN subscripted variable. Since there is no upper or lower case available in FORTRAN, subscripts are represented by enclosing them in parentheses. Now, the subscripted variable can be used in other statements; e.g., in a DØ loop as SUMX = SUMX + X (I).

20. The DO statement makes it possible to repeat the same operation, changing only the variable.
    (a) Control is shifted from the DO statement when the computations called for are completed, or by a GO TO or IF statement.
    (b) The general form of the DO statement:

$$\text{DO sn } i = m_1, m_2, m_3$$

where sn is a statement number, i is a nonsubscripted fixed-point variable, and m_1, m_2, and m_3 are each either unsigned fixed-point constants or nonsubscripted fixed-point variables. If m_3 is not stated, m_3 is understood to be 1. The DO statement tells the computer to execute repeatedly the statements which follow, up to and including the statement with the statement number sn. For the first iteration, the statements are executed with $i = m_1$. In each succeeding repetition, i is increased by the amount m_3. After the statements have been executed with i equal to the highest of the sequence of values which does not exceed m_2, control passes to the statement following the statement sn.

    (c) There are a number of rules concerned with the use of the DO statement. (1) The first statement in the range of a DO must be a statement that can be executed. (2) The range of one DO statement may contain another DO (called an inner DO). (3) The last statement in the range of a DO, with the exception of a GO TO or IF statement, may not cause a transfer of control. (4) No statement within the range of a DO may alter any of the indexing parameters of that DO. (5) Control must not transfer into the range of a DO from any statement outside its range.

21. The use of magnetic tape can greatly speed up the operation of a computer.
    (a) Magnetic tape will store intermediate results while the computer solves the remainder of the problem.
    (b) A read-input statement feeds data from the tape to the computer.

22. Open and closed functions are provided as part of the FORTRAN system.
    (a) Open functions are programmed each time they are needed.
    (b) Closed functions are stored, and used as needed.

23. The arithmetic statement function is used only in a particular program to perform repeated operations. This statement is limited in that it can compute only a single value.

24. Function and subroutine subprograms remove the limitations of the arithmetic statement function.
    (a) They are actually independent programs which have the advantage of dividing up a complex main program into workable segments.
    (b) Subroutines for common mathematical functions like sin, cos, log, square root, etc., are built into the FORTRAN system. Some of these obviate the use of tables for their evaluation.

An example is the square root function. SQRTF (X) computes the square root of X. It has one argument which is floating-point mode, and the function is floating point.

EXAMPLE: Y = SQRTF (A * X ** 2 − 4.0 * W)

Some other floating-point functions which are a part of the FORTRAN system are:

Name		Usage
ATAN	ATANF (X)	gives the principal value of arctan x in radians